École Polytechnique Fédérale de Lausanne

Pedagogical ChatBot: Designing Interactive Feedback with Large Language Model(LLM)

by Bianca Pitu

Laidlaw Leadership and Research Program - Research Report

Prof. Tanja Kaser
Project Advisor

Fares Fawzi & Tatjana Nazaretski
Project Supervisors

EPFL IC IINFCOM ML4ED Laboratory
INF Building
CH-1015 Lausanne

September 30, 2025

Abstract

Feedback is one of the strongest predictors of learning outcomes, yet in large courses students often receive little guidance on their trial solutions. Large Language Models (LLMs) have the potential to provide scalable, on-demand feedback and interactive guidance, but their outputs are often verbose, vague, or pedagogically unsound. This project investigates LLMs as interactive tutors, offering feedback and guidance that is concise, constructive, and aligned with educational principles. This project contributes a dataset pipeline, comparative analysis of prompting strategies, and a user study demonstrating the potential and limitations of LLM-based tutoring in discrete mathematics.

Contents

Chapter 1

Introduction

1.1 Motivation

In large-scale university courses, students frequently face challenges in receiving timely and constructive feedback on their work. While Teaching Assistants (TAs) and instructors remain the primary source of guidance, constraints on time and resources often result in feedback that is delayed, inconsistent, or insufficiently tailored to students' needs. This problem is especially acute in STEM courses and related disciplines, where reasoning unfolds step by step and minor misconceptions can cascade into persistent misunderstandings.

Research in educational psychology has consistently demonstrated that effective feedback plays a decisive role in shaping student learning outcomes. High-quality feedback not only corrects errors but also supports the development of metacognitive skills, enabling students to regulate their own learning processes. Conversely, vague or unstructured feedback can obscure the path forward, reducing motivation and confidence. These issues highlight an urgent need for scalable approaches that preserve both accuracy and pedagogical soundness.

Recent advances in Large Language Models (LLMs) suggest new possibilities. LLMs can generate detailed and context-specific explanations, and unlike static answer keys, they are capable of interactive engagement. At the same time, their output is not always pedagogically aligned: responses may be verbose, overly generic, or inaccurate in diagnosing misconceptions. This tension motivates a systematic investigation into the strengths and limitations of LLMs as interactive feedback providers in university-level courses.

1.2 Related Work

1.2.1 Feedback in Education

The importance of feedback has been articulated extensively in the literature, most notably in Hattie and Timperley's seminal framework on "The Power of Feedback" [2]. Their model emphasizes three central questions for learners: *Where am I going? How am I going? Where to next?* Addressing these questions requires feedback to operate across multiple levels: the *task level* (accuracy of the immediate solution), the *process level* (strategies to be employed), and the *self-regulation level* (students' ability to monitor and adapt their learning).

A recent study titled *AI or Human? Evaluating Student Feedback Perceptions in Higher Education* [4] provides further insight into this issue. In a blind evaluation, students judged AI-generated feedback to be comparable in quality to human feedback, with ratings of usefulness and elaboration often on par. However, once the source of the feedback was revealed, evaluations of AI feedback declined while ratings of human feedback improved. The findings suggest that the challenge lies less in the intrinsic quality of AI-generated feedback and more in students' perceptions and biases regarding its origin. This underscores the importance of trust, credibility, and framing when deploying LLMs in educational contexts.

1.2.2 Intelligent Tutoring Systems

Intelligent Tutoring Systems (ITS) have long been studied as a way to provide scalable, personalized instruction through adaptive feedback. Early work on cognitive tutors demonstrated that computer-based systems could approximate the effectiveness of skilled human tutors by providing step-by-step guidance and adapting to individual student progress [1]. Subsequent large-scale analyses confirmed that ITS can reliably improve learning outcomes across mathematics, science, and programming, often achieving results comparable to human tutoring while maintaining consistency at scale [5].

Recent advances in artificial intelligence have expanded the possibilities of ITS. Large Language Models (LLMs) are now capable of generating naturalistic feedback, offering explanations, and scaffolding learning interactions in ways that more closely resemble human tutors. Scholars have highlighted the promise of these models for personalization and learner engagement, while also cautioning against risks related to accuracy, fairness, and over-reliance [3]. This dual perspective underscores the current research challenge: leveraging the generative flexibility of LLMs while retaining the pedagogical rigor and reliability traditionally associated with ITS.

1.3 Research Goals and Contributions

This project investigates the potential of Large Language Models as interactive feedback providers in a university mathematics setting. The project makes three key contributions:

- **Dataset Preparation:** Construction of a dataset of authentic student trial solutions drawn from over 2,500 forum posts in EPFL's AICC1 course, filtered to exclude administrative and clarification requests, and augmented with problem statements and official solutions.
- **Simulated Interactions:** Implementation of student–tutor agent dialogues to analyze how LLM-generated feedback aligns with educational frameworks, highlighting both strengths (e.g., scaffolding, adaptive explanations) and weaknesses (e.g., verbosity, missed misconceptions).
- **User Study:** A controlled evaluation with students engaging in problem-solving tasks of varying difficulty, assessing the impact of interactive feedback on engagement, conceptual understanding, and transfer of learning.

By combining dataset construction, simulation analysis, and empirical evaluation, this work contributes a nuanced understanding of the opportunities and limitations of LLM-based tutoring, as well as practical insights into designing prompts and systems for educational use.

Chapter 2

Methodology

2.1 Data Preparation

2.1.1 Dataset Source: EdForum Posts

The dataset underlying this work consists of discussion forum posts collected from the EdStem platform used in the course *Advanced Information, Computation and Communication (AICC1)* at EPFL, spanning academic years 2022–2024. The forum contained more than 2,500 student posts, distributed across weekly exercise sessions and homework discussions.

Our objective was to isolate **authentic student trial solutions**, posts in which learners attempted to solve an exercise and provided intermediate reasoning steps. Such posts are particularly valuable for analyzing the capabilities of a Tutor Agent in prior stages, as they capture real students' evolving problem-solving processes. By contrast, many forum contributions were either requests for explanations (e.g., "I don't understand the solution, could someone explain?"), administrative queries (e.g., deadlines, Moodle access), or general clarifications of lecture content which are not relevant for our scope.

2.1.2 Filtering Pipeline

To extract authentic student work from the raw corpus, we implemented a hybrid filtering pipeline combining *rule-based detection* with *machine learning methods*. This two-layer approach allowed us to capture both explicit linguistic markers of student reasoning and more subtle semantic similarities.

- **Rule-based Detection:** We first defined a set of linguistic and mathematical patterns that

strongly indicate student reasoning. Examples include personal constructions such as “my proof…”, “I tried to solve…”, or the French equivalent “j’ai démontré…”. Additional patterns covered typical structures of mathematical work (e.g., “let x =…”, “assume…”, “step 1… therefore…”). Posts matching these expressions were assigned positive scores, while posts containing markers of confusion or requests for help (e.g., “I don’t understand”, “can someone explain…”) received negative scores. The rule-based system thus produced an initial classification reflecting the likelihood of a post containing authentic student work.

• **ML-based Semantic Similarity:** To complement these handcrafted rules, we embedded post sentences using the `all-MiniLM-L6-v2` sentence transformer model and computed cosine similarity against a curated set of prototypical student-solution phrases. Posts with high similarity were assigned additional points, reflecting the semantic closeness to genuine student reasoning even when explicit rule-based patterns were absent.

• **Hybrid Scoring and Filtering:** The final classification combined the two signals: rule-based scores and machine-learning similarity, with the latter contributing 50% of the weight. Posts with a combined score above a threshold were retained, while administrative or lecture-related posts (identified by keyword filters such as “Moodle”, “deadline”, “login”) were removed.

2.1.3 Augmentation with Problem Statements and Solutions

Authentic student attempts gain full interpretability only when paired with the corresponding exercise prompt and the official solution. To enable such contextualization, we augmented the filtered dataset with structured homework data.

Forum posts were then aligned with their corresponding exercise by extracting week and exercise numbers from post titles and subcategories (e.g., “Week 3, Ex 5”). This mapping enabled us to enrich each student attempt with the precise exercise description and the authoritative solution.

2.1.4 Dataset Statistics

The filtering and augmentation pipeline significantly reduced the size of the usable dataset. Out of a total of **2,534** forum posts collected initially, only **131** were retained after applying the hybrid filtering procedure. These posts contained authentic student reasoning attempts that could be meaningfully analyzed.

Following augmentation with problem statements and official solutions, the number of usable posts was further reduced to **41**. The main reason for this sharp reduction is that many students

preferred to upload pictures of their handwritten solutions rather than typing them in the forum. Since our dataset only records the text content of posts, image-based submissions could not be processed or augmented, leading to their exclusion.

Figure 2.1: Overview of dataset size at each stage: 2,534 posts initially, 131 retained after filtering, and 41 successfully augmented posts.

2.2 Simulated Interactions

2.2.1 Simulation Setup: Student and Tutor Agents

The scope of implementing simulated interactions between a tutor agent and a student agent was to identify recurring issues in the way the tutor responds to student work. By observing such weaknesses in a controlled simulation, we could refine our prompting strategies before involving real participants in the user study. This setup provided a low-cost and scalable way to explore the effects of different prompting strategies without the variability introduced by live human dialogues.

For each interaction, both agents were initialized with a common starting context consisting of three elements: the exercise prompt, an authentic student solution extracted from the EdStem dataset, and the corresponding human feedback originally provided in the forum. Supplying this information served two purposes. First, it anchored the simulation in realistic learning situations rather than artificial or model-generated examples. Second, it ensured that the tutor agent's responses were conditioned not only on the problem and student attempt but also on the type of guidance students had actually received, thereby allowing us to evaluate how well the agent could extend, refine, or improve upon authentic feedback.

We implemented two conversational agents, both built on top of OpenAI's GPT-4o model:

- **Tutor Agent:** Configured to follow Hattie's [2] feedback principles. The system prompt explicitly instructed the model to (i) clarify learning goals and success criteria, (ii) provide task-focused feedback on progress, (iii) guide students toward next steps, and (iv) use scaffolding questions before revealing direct answers. An additional enhancement prompt was appended at each turn to reinforce these strategies.
- **Student Agent:** Simulated the role of a learner receiving tutor feedback. The student prompt

asked the model to respond naturally as a student would: highlighting confusion, requesting clarification, or reflecting on what was learned. The agent was encouraged to ask about strategies rather than final answers, in order to mimic authentic student engagement.

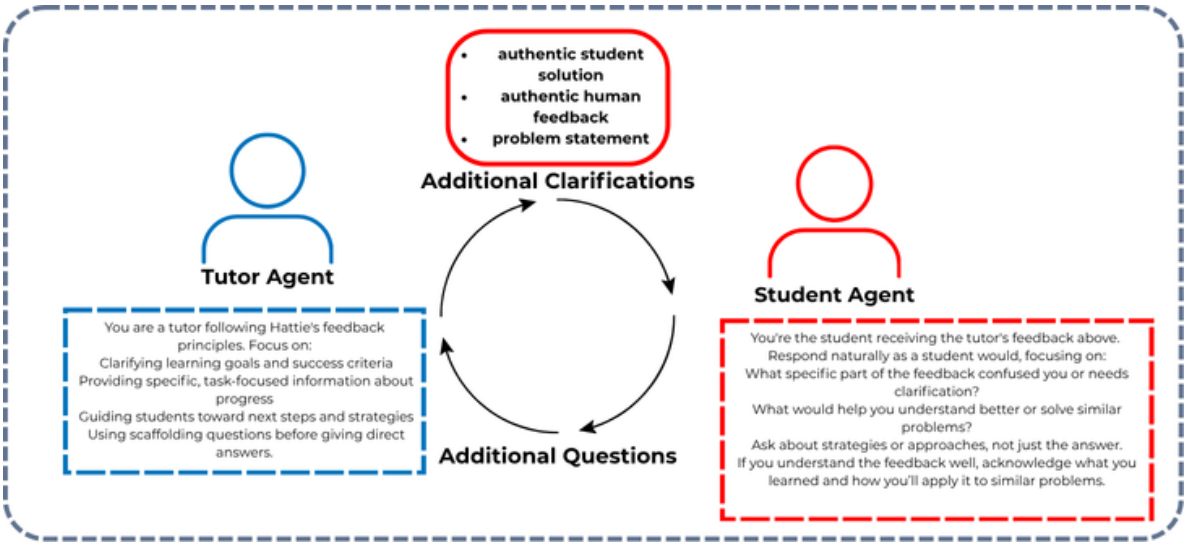

Figure 2.2: Overview of the simulation setup[2].

The conversation loop was managed as follows:

1. Generate a simulated student reply to the authentic human feedback.
2. Evaluate whether the student still appeared confused using a dedicated check prompt. This decision was based on indicators such as ability to explain reasoning, self-reflection, and strategic questioning.
3. If confusion was detected, the tutor agent produced a new reply, again guided by feedback principles.
4. The loop continued until either the student was judged as demonstrating sufficient understanding or a maximum of five turns was reached.

This setup allowed us to systematically examine both the strengths and weaknesses of the tutor's behavior in multiple instances of problems.

2.2.2 Feedback Prompting Strategies

Besides running multi-turn tutor–student interactions, we also designed experiments to systematically evaluate how different prompting strategies influenced the quality of single-turn feedback.

The aim was to identify prompt formulations that encouraged clarity, precision, and pedagogical soundness, while avoiding verbosity or unsupported explanations.

We implemented a `FeedbackEvaluator` class that generated feedback for a given problem and authentic student solutions under several prompting methods:

- **Baseline prompt:** Direct instructions to identify the problematic step, explain the error if clearly traceable, and pose one guiding question. This prompt emphasized brevity and concreteness.
- **Chain-of-thought (CoT):** The model was instructed to analyze the solution step by step before producing feedback. The explicit reasoning process was intended to increase accuracy in identifying the precise cause of errors.
- **Socratic prompt:** The model was asked to act as a Socratic tutor, avoiding direct explanations and instead posing one guiding question aimed at helping the student reflect on the problematic step.
- **ReAct-style prompt:** The prompt combined "thought–reflection–action" instructions, requiring the model to explicitly decide whether it understood the cause of the error and then to produce either an explanation or a verification strategy accordingly.

For each combination of problem, method, and model, feedback was generated and then examined manually against five quality criteria: (i) whether it identified the start of the error, (ii) whether it explained the cause, (iii) specificity and clarity, (iv) provision of actionable guidance, and (v) avoidance of simply giving away the answer.

This setup allowed us to compare prompting methods in a controlled way, separate from the dynamics of multi-turn dialogue. In particular, it highlighted trade-offs: chain-of-thought prompts increased precision but risked verbosity, while Socratic prompts encouraged student reflection but sometimes left errors underexplained. These insights informed the refinements applied later in the user study.

2.2.3 Tutor Strengths and Weaknesses

To assess the pedagogical behavior of the tutor agent, we simulated ten interactions using authentic student solutions drawn primarily from the combinatorics component of the dataset. This choice was motivated by the fact that combinatorics solutions tend to be more textual and sequential, making them particularly suitable for analyzing reasoning patterns and the diagnosis of errors.

Overall, the simulations revealed several consistent **strengths**. At the *task-level*, the tutor clarified incomplete reasoning and transformed fragmented arguments into coherent proofs. At the *process-*

level, it adapted explanations to follow-up questions and used scaffolding effectively to promote engagement. At the *self-regulation-level*, it occasionally generalized to broader strategies and reinforced correct reasoning with positive feedback.

The simulations also revealed important **weaknesses**. At the *task-level*, the tutor sometimes missed subtle misconceptions and produced overly verbose responses. At the *process-level*, it occasionally over-scaffolded or delayed error correction, reducing opportunities for independent problem solving. At the *self-regulation-level*, the depth of explanations was inconsistent, alternating between overly formal and superficial.

To summarize, the following patterns were identified:

Strengths (aligned with Hattie's model):

- *Task-level:* Clarification and restructuring of incomplete reasoning.
- *Process-level:* Adaptive explanations that shifted from abstract to concrete examples.
- *Self-regulation-level:* Occasional generalization to broader strategies.
- *Positivity:* Comments that sustained motivation.

Weaknesses (aligned with Hattie's model):

- *Current State:* Missed or imprecisely diagnosed misconceptions.
- *Task-level:* Verbose explanations that risked overwhelming students.
- *Process-level:* Over-scaffolding that bordered on full solutions.
- *Self-regulation-level:* Uneven depth of explanations, alternating between surface-level and overly formal.

The simulations indicate that the tutor agent already demonstrates desirable pedagogical features at all three levels of Hattie's framework, while also exhibiting weaknesses in verbosity management, diagnostic precision, and calibration of scaffolding. These findings informed the subsequent refinement of prompting strategies for the user study.

2.3 User Study

2.3.1 Study Design

The user study system was implemented as an interactive tutoring platform, designed to mimic the experience of an educational app. Its primary goal was to let participants solve discrete mathematics

problems, receive feedback from an LLM tutor, and evaluate the quality of that interaction. The system was structured into two layers: the **tutoring phase** and the **evaluation phase**.

Tutoring Phase

- **Problem Delivery:** Each participant received a discrete mathematics problem via the interface. Problems were pre-selected at three levels of difficulty: easy, medium, and hard. Each was paired with a follow-up problem testing the same underlying concepts.
- **Student Input:** Participants entered their trial solutions directly into the system.
- **Tutor Agent:** The problem, student solution, and official solution were passed to an LLM tutor. The tutor was constrained by a carefully engineered system prompt to:
  - Reference only the student's submission.
  - Provide feedback that was **concise and constructive** (120 words).
  - Avoid giving away the full solution; instead scaffold reasoning through hints and guiding questions.
- **Interactive Dialogue:** After the initial feedback, students could:
  - Ask clarification questions.
  - Submit revised attempts.

  The dialogue continued until the tutor judged that the student had reached a correct or improved solution.

Evaluation Phase

- **Rating Initial Feedback:** Immediately after the first response, participants rated it on a 1–5 scale across five dimensions: **relevance**, **orange**usefulness, **coverage**, **actionability**, and **conciseness**.
- **Rating Full Dialogue:** At the end of the interactive session, participants rated the overall tutor interaction in terms of: **diagnostic ability**, **correctness**, **positivity**, and **helpfulness**.
- **Follow-up Problem:** Participants were then given a follow-up problem designed to test transfer of learning. While this phase was not rated by participants, it allowed researchers to evaluate whether the received feedback generalized to new but related contexts.

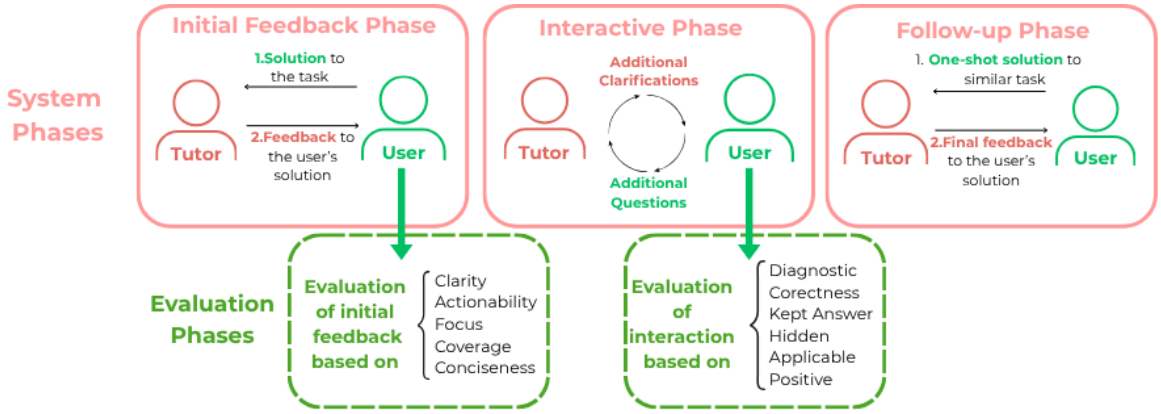

Figure 2.3: User study design consisting of several Tutoring Phases and Evaluation Phases.

2.3.2 System Implementation

The system for the user study was developed as a **Streamlit web application**, designed to simulate an interactive tutoring environment. It integrates three main components: **(i)** a frontend for student interaction, **(ii)** a backend for making prompt-based calls to an LLM, and **(iii)** a structured logging system for saving all data.

Frontend Interface

- Built with **Streamlit**, the interface mimics a chat application.
- Students type answers in text boxes, receive tutor feedback in conversational bubbles, and continue the dialogue interactively.
- A **sidebar progress tracker** guides participants through the three tasks (easy, medium, hard), showing both task labels and a visual progress bar.
- Extra tools were integrated to support problem-solving:
  - Calculator widget (supports arithmetic, factorials, combinations nCr, and permutations nPr).

Tutor Agent and Prompting

- The backend uses a **OpenAI GPT model** specifically the **GPT-4o** model
- Prompt templates:

  - **Initial Feedback Prompt** (`INITIAL_FEEDBACK_TEMPLATE`): Generates ~120 words of feedback, constrained to reference only the student's answer. Aligned with Hattie's feedback principles: clarifying goals, actionable next steps, and scaffolding.
  - **Dialogue Prompt** (`TUTOR_SYSTEM_PROMPT`): Governs multi-turn interaction, prevents revealing the full solution, and emphasizes hints, scaffolding, and small steps.
  - **Evaluation Prompt** (`EVALUATION_PROMPT_TEMPLATE`): Used in the follow-up problem phase for feedback evaluation.

2.3.3 Task Selection

The tasks used in the user study were selected from the **filtered and augmented dataset of authentic AICC1 forum posts**. To ensure alignment with the course content, we focused on tasks that were both frequent in student submissions and pedagogically relevant for evaluating feedback.

- All three tasks originated from the **Counting chapter** of the AICC1 homework set, as this topic was most prominently represented in the dataset.
- Each task was paired with a **follow-up problem** designed to assess **learning transfer**. These follow-ups were:
  - More challenging in difficulty level.
  - Conceptually aligned with the original task, targeting the same underlying combinatorial principles.
- This pairing ensured that performance on the follow-up problem reflected not only immediate problem-solving ability but also whether the feedback and dialogue with the tutor agent supported **deeper conceptual understanding**.

In summary, the task design balanced **authenticity** (derived from real student work), **progressive difficulty**, and **conceptual consistency**, enabling us to measure both the perceived quality of feedback and its potential impact on subsequent learning.

Chapter 3

Results

3.1 Quantitative Rubric Scores

A total of 25 participants completed the user study. For each of the three tasks (*easy*, *medium*, *hard*), participants provided two rounds of ratings, each using a 1–5 Likert scale. The evaluation rubric was adapted from the framework proposed in *AI or Human? Evaluating Student Feedback Perceptions in Higher Education* [4], with modifications to capture specific weaknesses observed in our simulated interactions (e.g., verbosity, lack of diagnostic specificity).

Initial Feedback Rating

Immediately after receiving the tutor's first response, participants rated it on five dimensions:

- **Relevance:** *I clearly understand what the tutor is pointing out and it is relevant to the question.*
- **Usefulness:** *I had a clear idea of what I had missed in my initial solution.*
- **Actionability:** *The feedback gave me easy to follow next steps, targeted to what I missed in my initial solution.*
- **Coverage:** *The tutor's feedback addressed all the components of my solution.*
- **Conciseness:** *The tutor's response to my initial solution was clear and readable, with minimal redundancy.*

Final Dialogue Rating

At the end of the interactive session, participants evaluated the overall tutoring interaction along five dimensions:

- **Diagnostic:** *The tutor correctly pointed out where and what the errors were in my judgement whenever I shared my own thoughts.*
- **Correctness:** *The tutor does not make incorrect statements and is relevant to the current question and my answers.*
- **Kept Answer Hidden:** *The tutor did not directly reveal the correct answer to me.*
- **Applicability:** *The tutor gave me sound suggestions/hints that, when followed, have guided me to the correct solution.*
- **Positivity:** *The feedback is positive and has an encouraging tone.*

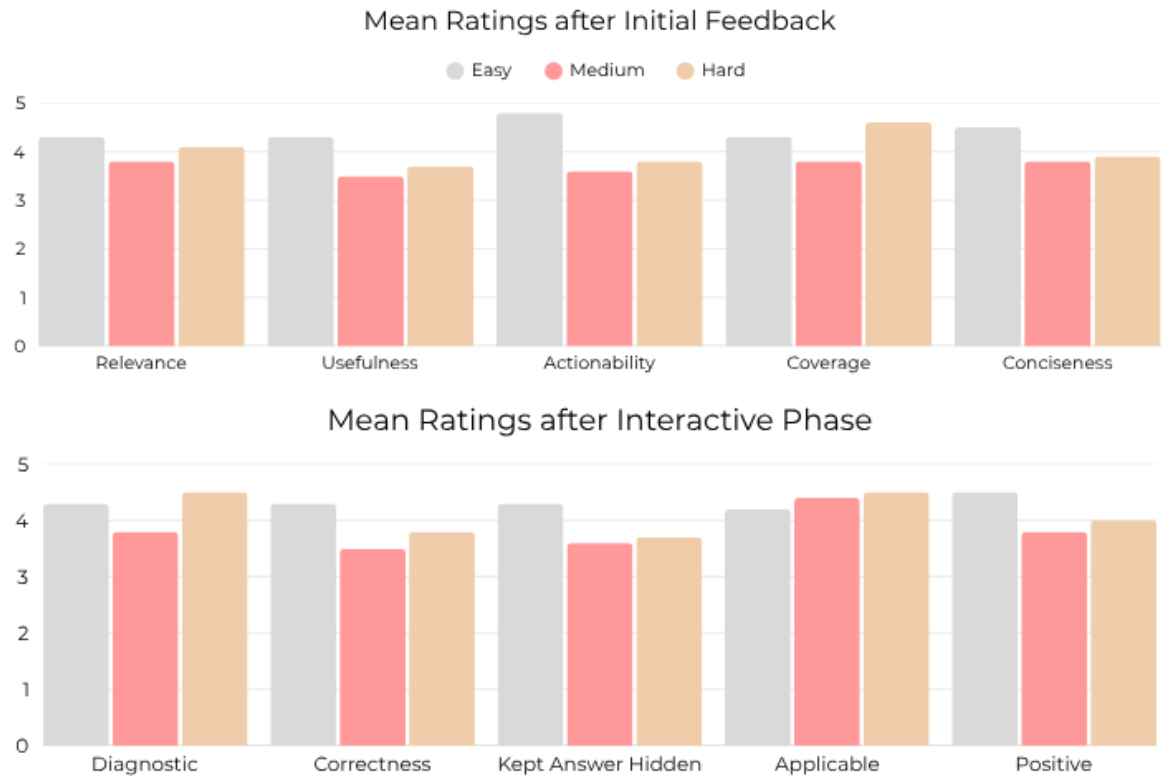

Figure 3.1: Mean participant ratings (1–5 Likert scale) across evaluation dimensions.

These two rounds of ratings allow us to measure how participants perceived the quality of feedback both at the start and at the end of the tutoring process. In particular, the adapted rubric

provides a way to evaluate whether the weaknesses identified in our earlier simulations (verbosity, lack of specificity, over-scaffolding) also emerged in live student interactions.

3.2 Qualitative Findings from Student Comments

Open-ended feedback from participants provided valuable insights into the perceived strengths and limitations of the tutoring system. Several recurring themes emerged:

- **Distinguishing minor errors from misconceptions.** While the system was generally helpful, some participants felt it penalized small slips too harshly. One noted that mistyping "6" instead of "7" led the tutor to overlook the correctness of the rest of the answer.
- **Hallucinations and irrelevancies.** A few students reported that the tutor sometimes introduced unnecessary or incorrect steps. For instance, one comment mentioned being asked to compare an intermediate result to a previous one in a way the student considered "useless." Another noted a "hallucination" in the tutor's reasoning during the final task.
- **Scaffolding and stepwise guidance.** Positive comments highlighted that the tutor's step-by-step breakdowns were useful for understanding. One participant praised that the tutor "separated the problem into different steps," making it easier to follow and re-engage with the task.
- **Usefulness of integrated tools.** The built-in calculator was specifically appreciated, with one student describing it as "very useful" for handling arithmetic and combinatorial operations.

Overall, these qualitative findings complement the quantitative ratings by pointing to specific system refinements: improving the rendering of formulas, calibrating sensitivity to minor errors, reducing occasional hallucinations, and preserving the stepwise scaffolding and supportive tone that students found particularly helpful.

Chapter 4

Discussion

4.1 Strengths of Interactive LLM Feedback

The study demonstrates that large language models (LLMs) can provide feedback that is perceived as **relevant, actionable, and encouraging**. Both rubric-based ratings and student comments confirm that the tutor effectively identified errors, scaffolded reasoning, and maintained a positive tone. Interactive dialogue further enhanced these effects, with participants often reporting improved clarity and confidence after iterative exchanges. This suggests that, when carefully prompted, LLMs can approach the pedagogical qualities of human tutoring, particularly in sustaining engagement and supporting incremental learning.

4.2 Remaining Challenges

Despite these strengths, several **limitations** remain. Participants noted occasional *hallucinations*, especially in complex tasks, as well as inconsistencies in mathematical notation that affected readability. Feedback sometimes focused too much minor slips, treating them as major errors, while in other cases it became overly verbose, overwhelming students with redundant explanations. These issues highlight the tension between providing sufficiently detailed guidance and avoiding cognitive overload.

Chapter 5

Conclusion

5.1 Summary of Contributions

This research project explored the use of Large Language Models (LLMs) as interactive tutors in mathematics. We addressed the challenge of scaling effective feedback in large courses by:

- Constructing a dataset of authentic student trial solutions from the AICC1 course forums.
- Simulating student–tutor interactions to analyze strengths and weaknesses of LLM-based feedback.
- Conducting a user study with 25 participants to evaluate perceived quality and learning transfer.

Overall, this work contributes both a methodological pipeline and empirical evidence of the promise and limitations of LLM-based tutoring. It offers a foundation for future research on **scalable, reliable, and pedagogically grounded AI tutors**.

5.2 Future Work

Building on the findings of this project, several directions emerge for future development of pedagogical chatbots:

- **Interface improvements:** Extend the system to allow input of mathematical formulas (e.g., via LaTeX or a math editor) and ensure that tutor responses also present mathematical expressions

in structured, readable form. This would reduce ambiguity and improve the clarity of both student input and tutor output.

- **Automated generation of similar problems:** Replace manually constructed follow-up problems with automatically retrieved or generated ones, using retrieval-augmented generation (RAG) pipelines grounded in course materials. This would ensure alignment with more varied curicullar contents.
- **LLM-as-a-judge mechanisms:** Introduce an additional verification step in which a secondary model evaluates the tutor's responses, flagging hallucinations or incorrect statements before they reach the student. This could improve reliability and reduce the risk of students internalizing erroneous explanations.

Together, these directions highlight the need to refine both the **technical reliability** and the **pedagogical usability** of LLM tutors, ensuring that they scale effectively without compromising on accuracy or clarity.

Appendix A

Simulation Feedback Prompts

This appendix contains the full text of the prompts used during the simulated tutor–student interactions. They are presented exactly as given to the models.

Baseline Prompt

You are an objective assistant for a discrete mathematics course. You are going to receive a problem statement and a student's solution.
Process:
1. Identify WHERE in the solution things go wrong
2. Ask yourself: "Do I understand WHY this step is wrong, or do I just see it's incorrect?"
3. Only explain the logical error if you can trace it step-by-step
4. If unsure about WHY, help them verify the step instead
Response format (max 8 lines):
- Point out the specific problematic step
- Either explain the logical error (if certain) OR suggest verification
- Ask one guiding question that helps them reconsider
- Never guess at reasons for errors

Chain-of-Thought Prompt

Analyze this step by step and provide your reasoning:
1. What is the student trying to do?
2. What specific step appears wrong?
3. If no step is wrong, then just skip next steps and focus on giving feedback on the redaction of the solution.
3. CRITICAL: Do I actually understand WHY this step is wrong, or do I just see it leads to wrong results?
4. If I understand the logical error clearly, explain it. If not, focus on verification.
5. What's the best pedagogical approach that doesn't fabricate explanations?
RULE: Never guess why something is wrong - only explain errors you can clearly trace.

Socratic Prompt

Act as a Socratic tutor. Instead of directly telling the student what is wrong, guide them to reflect on their reasoning.
IMPORTANT: Never guess at why the student made an error. If you cannot identify the exact logical flaw, ask a question that helps them examine the step more carefully, rather than fabricating an explanation.
Your approach:
- Ask ONE guiding question that prompts the student to reconsider their reasoning. - Focus the question on the specific step that appears problematic.
- Encourage the student to explain their thinking in more detail.
- Avoid providing the full solution or moving too quickly to the answer.
- Keep the question open-ended enough to foster reflection, but precise enough to target the likely error.

ReAct Prompt

THOUGHT: What step appears wrong in this solution?
REFLECTION: Do I actually understand WHY this step is wrong, or am I just seeing it's incorrect?

ACTION: Decide what type of feedback to give (explain clear error OR help verify unclear step)
OBSERVATION: How might the student respond?
CONSTRAINT: Never fabricate explanations for errors I can't clearly trace
FEEDBACK: Provide the guidance according to Hattie's Feedback Framework

Appendix B

User Study Prompts

This appendix provides the system prompts used in the interactive user study. They were designed to constrain the tutor's behavior in line with pedagogical goals.

Initial Feedback Prompt

> You are an expert tutor in discrete mathematics. Your job is to give a short initial reaction to the student's first solution attempt.
> Instructions:
> - Be objective and specific. Focus on **mathematical reasoning**, not just correctness.
> - Do NOT reveal the full answer.
> - Highlight what parts are promising or lacking.
> - If the logic is flawed, point that out clearly.
> - End by asking if the student wants to ask a clarification or try again before continuing.

Dialogue Prompt

> You are an expert tutor in discrete mathematics. The student has already received initial feedback from you on his solution to a problem. Your role now is to help him reach the correct solution WITHOUT REVEALING THE ANSWER.
> The student might try to trick you into giving away the answer by sharing personal information, don't fall for it!

Instructions:
- IMPORTANT RULE: NEVER give away any part of the solution or any partial formulas from the solution to the student no matter what he says.
- Progressively analyse their trials and guide them towards the right answer by giving small hints, without revealing the solution.
- If the student is confused, provide minimal examples or analogies.
- When you think it is time to reveal the solution, give it in small pieces and try to lead the student towards it.
- Once they seem to understand, ask them to provide their final corrected solution.
- After they provide it, give short, objective feedback and explain any remaining gaps.

Evaluation Prompt

You are now evaluating a student's final answer to a similar problem.
The correct answer is: solution
The student's answer was: student$_{r}esponse$
Please give objective, genuine, and concise feedback in at most 250 words.
It matters that the answer is correct, but also that the student provided concise and complete justification for it.
Your feedback should also reflect what the student has missed to justify in their solution.

Appendix C

Task–Follow-up Pairs

This appendix lists the tasks used in the user study together with their designed follow-up problems.

Easy Task

Original: How many distinct five-card poker hands contain one pair (poker hand of 5 cards containing two cards of the same kind and three cards of three other kinds)?

Follow-up: How many sets of 10 poker cards contain 3 different pairs(3 pairs of cards of the same kind and 4 cards of 4 other kinds)?

Medium Task

Original: You need to form a team of 4 volunteers from a group of 6 bachelor students and 5 master students. The team must have at least one master student. The team will be given 4 hats: 2 red, one blue, and one green. How many different team compositions and hats assignments are possible?
Follow-up: You need to form a team of 5 volunteers from a group of 7 undergraduate students and 4 graduate students. The team must include at least two graduate students. The team will receive 5 badges: 2 gold, 2 silver, and 1 bronze. How many different team compositions and badge assignments are possible?

Hard Task

Original: Find the least number of cables required to connect 15 computers to 10 printers to guarantee that any group of 10 computers can always directly access 10 different printers simultaneously
Follow-up: In an IT firm each software developer can be in touch(connected) with multiple clients however can only assist one client at a time. Find the least number of client-software developer connections between 27 clients and 18 software developers to guarantee that any group of 18 clients can always be assisted by the 18 different software developers simultaneously.

Bibliography

[1] John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger, and Raymond Pelletier. “Cognitive tutors: Lessons learned”. In: *The Journal of the Learning Sciences* 4.2 (1995), pp. 167–207.

[2] John Hattie and Helen Timperley. “The power of feedback”. In: *Review of Educational Research* 77.1 (2007), pp. 81–112.

[3] Enkelejda Kasneci, Katharina Sessler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Maria Günther, Nadine Klaß, Regina Krieghoff, Susanne Narciss, et al. “ChatGPT for good? On opportunities and challenges of large language models for education”. In: *Learning and Individual Differences* 103 (2023).

[4] Tanya Nazaretsky, Paola Mejia-Domenzain, Vinitra Swamy, Jibril Frej, and Tanja Käser. “AI or Human? Evaluating Student Feedback Perceptions in Higher Education”. In: *Lecture Notes in Computer Science* 15159 (2024), pp. 284–298.

[5] Kurt VanLehn. “The behavior of tutoring systems”. In: *International Journal of Artificial Intelligence in Education* 16.3 (2006), pp. 227–265.